



Задача А. Юный Сальвадор

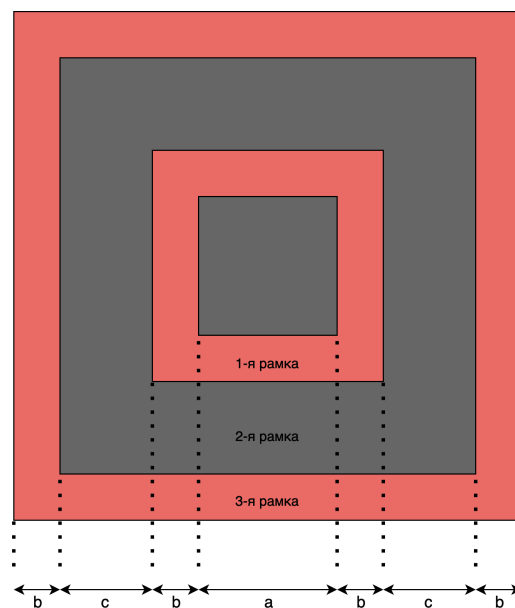
Ограничение по времени: 1 секунда

Ребенок взял бесконечный клетчатый лист и начал закрашивать на нем клетки черным и красным цветами.

Для начала он нарисовал черный квадрат со стороной длины a . Затем он k раз добавил к текущему рисунку «рамку» вокруг него по следующим правилам:

- каждый нечетный раз он отступал b клеток от каждой стороны текущего рисунка и закрашивал полученную рамку красным;
- каждый четный раз он отступал c клеток от каждой стороны рисунка и закрашивал полученную рамку черным.

Пример для $a = 3$, $b = 1$, $c = 2$ и $k = 3$ виден на рисунке ниже:



Определите, сколько клеток будет закрашено черным и красным цветом после того, как рисунок будет закончен.

Формат входных данных

В единственной строке ввода через пробел перечислены четыре целых числа a , b , c и k — длина стороны исходного черного квадрата, длины отступов для красных и черных рамок и количество рамок, нарисованных вокруг исходного квадрата ($0 \leq a, b, c \leq 10^4$; $0 \leq k \leq 10^5$).

Формат выходных данных

Выведите через пробел сначала число черных клеток, затем число красных клеток.

Система оценки

Каждая группа тестов будет оцениваться только если предварительно были пройдены необходимые группы тестов, и баллы начисляются в случае, если все тесты группы пройдены. Тесты из условия не оцениваются. Все тесты разбиты на группы со следующими ограничениями:



Подзадача	Доп. ограничения	Необх. подзадачи	Баллы
1	$k \leq 1$		10
2	$k \leq 2$	1	13
3	$a = 0$		14
4	$c = 0$		14
5	$a, b, c \leq 4; k \leq 20$		12
6	$a, b, c \leq 4; k \leq 800$	5	17
7	нет	1 – 6	20

Примеры

стандартный ввод	стандартный вывод
3 1 2 3	65 56
1 1 1 6	97 72



Задача В. Патерны Даттона

Ограничение по времени: 1 секунда

Хомяку Даттону из Татарстана подарили строку s длины n .

Ему стало интересно, сколько существует подстрок строки s вида $a_1(k)a_2(i)a_3(j)$, где a_1, a_2 и a_3 обозначают строчные латинские буквы, а числа в скобках после символа — то, сколько раз этот символ повторяется, но важно, чтобы $a_1 \neq a_2$ и $a_2 \neq a_3$. Например, $i(3)n(2)o(1)$, — это описание строки $iiinno$.

Формат входных данных

Первая строка входных данных содержит целое число n — длина строки s .

Вторая строка содержит строку s длины n , состоящую из строчных латинских букв.

В третьей строке записаны числа k, i, j , описывающие количество символов a_1, a_2, a_3 в подстроках соответственно.

Формат выходных данных

Выведите количество подстрок, представимых в виде $a_1(k)a_2(i)a_3(j)$.

Система оценки

Каждая группа тестов будет оцениваться только если предварительно были пройдены необходимые группы тестов, и баллы начисляются в случае, если все тесты группы пройдены. Тесты из условия не оцениваются. Все тесты разбиты на группы со следующими ограничениями:

Подзадача	Ограничения	Необх. подзадачи	Баллы
1	$n \leq 100$	—	30
2	$n \leq 1000$	1	30
3	$n \leq 10^6$	1, 2	40

Пример

стандартный ввод	стандартный вывод
10 llvvyuwww 2 2 2	2



Задача С. Коровы проверяют Даттона

Ограничение по времени: 2 секунды

Ранчо Даттона — это бесконечное клетчатое поле, на каждой клетке которого изначально записан 0.

Коровы на ранчо Даттона решили проверить своего хозяина, поэтому они решили написать инструкцию для коровы Бет, о том, как она должна передвигаться по полю.

Бет стартует в клетке $(0; 0)$. Инструкция представляет из себя строку s , состоящую из n символов. Каждый символ — это одна команда. Существует четыре основных команды, доступных для выполнения:

- U — перейти из клетки $(x; y)$ в $(x; y + 1)$;
- D — перейти из клетки $(x; y)$ в $(x; y - 1)$;
- L — перейти из клетки $(x; y)$ в $(x - 1; y)$;
- R — перейти из клетки $(x; y)$ в $(x + 1; y)$.

На первом шаге Бет записывает число 1 в клетке $(0; 0)$, на втором шаге — число 2 в клетке, в которой окажется после первой команды и т.д. Если она окажется в клетке, в которой уже была до этого, то перезапишет в ней число на новое.

После того, как Бет пройдет по полю, коровы зададут q запросов. Существует два вида запросов:

- $1 x$ — нужно сказать сумму всех чисел в строчке x ;
- $2 y$ — нужно сказать сумму всех чисел в столбце y .

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 10^5$) — количество команд.

Во второй строке записана строка s из n символов U, D, L или R , обозначающих команды.

В третьей строке дано целое число q ($1 \leq q \leq 10^5$) — количество запросов.

В следующих q строках содержатся запросы вида « $1 x$ » и « $2 y$ », требующие ответ - сумму чисел в строке или столбце соответственно.

Формат выходных данных

Необходимо вывести q чисел — ответов на запросы, каждое на отдельной строке.

Система оценки

Каждая группа тестов будет оцениваться только если предварительно были пройдены необходимые группы тестов, и баллы начисляются в случае, если все тесты группы пройдены. Тесты из условия не оцениваются. Все тесты разбиты на группы со следующими ограничениями:

Подзадача	Ограничения	Необх. подзадачи	Баллы
1	$n \leq 100; q \leq 100$	—	30
2	$n \leq 500; q \leq 10^5$	1	30
3	$n \leq 10^5; q \leq 10^5$	1, 2	40



Пример

стандартный ввод	стандартный вывод
10	24
DDURULLDRU	0
5	11
1 0	25
2 2	0
1 1	
2 0	
1 -3	



Задача D. Забор

Ограничение по времени: 1 секунда

Даттон узнал, что всего было установлено n заборов. Каждый забор описывается как отрезок $(l; r)$.

Датон, хомяк-математик, дал Джейми n отрезков, обозначенных как (l_i, r_i) , где $1 \leq l_i \leq r_i \leq m$. Джейми хочет добавить один новый отрезок (l, r) , где $1 \leq l \leq r \leq m$.

Однако Датон поставил условие: новый отрезок (l, r) можно добавить только в том случае, если ни один из существующих отрезков (l_i, r_i) не будет полностью находиться внутри нового отрезка (l, r) .

Вам нужно определить количество различных способов выбрать такой отрезок (l, r) , который удовлетворяет указанному условию.

Формат входных данных

В первой строке записаны целые числа n и m ($1 \leq n, m \leq 10^5$) — количество отрезков и их максимально возможная координата соответственно.

В следующих n строках описываются сами отрезки целыми числами l_i и r_i ($1 \leq l_i \leq r_i \leq m$) — левой и правой границей отрезка соответственно.

Формат выходных данных

Вывести одно число — количество различных способов выбрать новый отрезок.

Система оценки

Каждая группа тестов будет оцениваться только если предварительно были пройдены необходимые группы тестов, и баллы начисляются в случае, если все тесты группы пройдены. Тесты из условия не оцениваются. Все тесты разбиты на группы со следующими ограничениями:

Подзадача	Ограничения	Необх. подзадачи	Баллы
1	$n \leq 100; m \leq 100$	—	30
2	$n \leq 10^5; m \leq 100$	1	35
3	$n, m \leq 10^5;$	1, 2	35



Примеры

стандартный ввод	стандартный вывод
2 10 1 2 3 4	32
3 10 1 5 5 10 2 5	40
4 10 1 5 5 10 2 5 3 7	37
5 10 1 5 5 10 2 5 3 7 7 9	31



Задача Е. Подвалы и Математики

Ограничение по времени: 1 секунда

Разработчик VK Рустем очень любит играть в настольные игры, особенно в перерывах между созданием новых функций для своих сервисов. В этот раз он решил устроить сражение в игре "Подвалы и Математики". Рустем собирает друзей сразиться в игре Подвалы и Математики.

Для каждой партии необходимо создавать новое поле, которое представляет собой правильный k -угольник. Рустем знает размеры прямоугольного стола длиной n и шириной m , на котором будет проходить их игра.

Его поле должно полностью помещаться на столе и при этом быть максимально возможным по площади. Но есть одна особенность: чтобы поле не двигалось по столу, его левый нижний угол прикрепляют к левому нижнему углу стола так, чтобы поле не выходило за границы стола. Помогите Рустему найти максимально возможную площадь поля или сказать, что такого поля быть не может. Если поле сделать невозможно, выведите число -1 .

Формат входных данных

В единственной строке записаны целые числа n и m ($1 \leq n, m \leq 1000$), обозначающие длину и ширину стола соответственно, а также целое число k ($3 \leq k \leq 10^5$) — количество вершин правильного многоугольника, который является полем.

Формат выходных данных

Вывести вещественное число — площадь поля, соответствующего условиям, с точностью не менее 10^{-6} . То есть ваш ответ должен отличаться от необходимого не более чем на 10^{-6} . Если ответа не существует, необходимо вывести только число -1 .

Система оценки

Каждая группа тестов будет оцениваться только если предварительно были пройдены необходимые группы тестов, и баллы начисляются в случае, если все тесты группы пройдены. Тесты из условия не оцениваются. Все тесты разбиты на группы со следующими ограничениями:

Подзадача	Ограничения	Необх. подзадачи	Баллы
1	$k = 4$	—	15
2	$n = m; 3 \leq k \leq 4$	1	30
3	$k = 3$	—	35
4	$k \leq 10^5$	1, 2, 3	20

Примеры

стандартный ввод	стандартный вывод
10 11 4	100.000000000000000000000000000000
10 10 3	46.41016151377545652726



Задача F. Ночной жор Даттона

Ограничение по времени: 1 секунда

У хомяка Даттона n переключателей, контролирующих свет. Он боится темноты, поэтому просит вас написать ему программу, которая поможет ему.

Его дом выглядит как коридор, состоящий из n комнат, соединенных в ряд. В каждой комнате есть лампочка, которая изначально может быть включена «1» или выключена «0».

Начальные состояния лампочек описаны массивом L . Даттон часто просыпается ночью, чтобы пойти поесть, поэтому перед сном хочет переключить все лампочки так, чтобы он мог пройти по коридору из своей спальни на кухню так, чтобы в каждой комнате, по которой придется пройти, лампочка была включена.

Так как он хочет сэкономить на электричестве, то во всех остальных комнатах свет должен быть выключен.

Кухня находится в комнате с номером A . Если $A = 0$, то он может попасть на кухню напрямую из спальни, и все лампочки в коридоре должны быть выключены. Если $A > 0$, то в комнатах с номерами с 1 по A лампочки должны остаться включенными.

Переключать лампочки он может следующим образом: выбрать k подряд идущих комнат коридора, во всех комнатах из k выбранных, если лампочки были включены, они станут выключенными, и наоборот. То есть «0» будут изменяться на «1», а «1» на «0».

Даттон устал после тяжелого рабочего дня, поэтому хочет сделать как можно меньше таких переключений.

Ваша задача — написать программу, которая должна найти минимальное количество действий, которые Даттону нужно сделать или сказать, что его просьбу выполнить невозможно.

Формат входных данных

В первой строке содержатся числа n ($1 \leq 10^5$) — количество комнат в коридоре и k ($1 \leq n$) — количество подряд идущих лампочек, которые переключаются за один ход.

Во второй строке записан массив L размера n — состояния лампочек в начальный момент, где лампочка в комнате с номером i включена «1» или выключена «0».

В третьей строке единственное число A ($0 \leq A \leq n$) — номер комнаты, в которой находится кухня.

Формат выходных данных

Выведите одно число — минимальное количество действий, необходимое для переключения лампочек вышеописанным способом. Если при любом количестве действий и любой их последовательности так сделать невозможно, то выведите -1 .

Система оценки

Каждая группа тестов будет оцениваться только если предварительно были пройдены необходимые группы тестов, и баллы начисляются в случае, если все тесты группы пройдены. Тесты из условия не оцениваются. Все тесты разбиты на группы со следующими ограничениями:

Подзадача	Ограничения	Необх. подзадачи	Баллы
1	$k = 1$	—	10
2	$n \leq 1000; A = 0$	—	20
3	$n \leq 10^5; A = 0$	2	30
4	$n \leq 10^3$	1	20
5	$n \leq 10^5$	1, 2, 3, 4	20



Примеры

стандартный ввод	стандартный вывод
10 3 1 0 0 0 0 0 1 1 1 0 1	1
10 3 1 0 0 0 1 1 0 1 1 0 1	2



Задача G. Ильнур и массив

Ограничение по времени: 1 секунда

Ильнуру дан массив чисел длины n , i -е число массива — a_i . Для двух чисел с индексами i и j Ильнур определяет следующую величину: $P(i, j) = \lceil \sqrt{a_i \cdot a_j} \rceil$, где $\lceil \cdot \rceil$ означает округление вверх.

Ильнур называет массив хорошим, если в нем нет двух различных пар чисел с одинаковой величиной P . Помогите Ильнуру определить, является ли его массив хорошим?

Формат входных данных

В первой строке дано единственное целое положительное число n — размер массива. Во второй строке записаны n целых положительных чисел a_1, \dots, a_n .

Каждая группа тестов будет оцениваться только если предварительно были пройдены необходимые группы тестов, и баллы начисляются в случае, если все тесты группы пройдены. Все тесты разбиты на группы со следующими ограничениями:

Подзадача	Ограничения	Необх. подзадачи	Баллы
1	$n \leq 10; a_i \leq 100$	—	10
2	$n \leq 10^3; a_i \leq 10^6$	1	15
3	$n \leq 10^6; a_i \leq 10^6$	1, 2	75

Формат выходных данных

Выведите YES, если массив Ильнура является хорошим, NO иначе.

Примеры

стандартный ввод	стандартный вывод
3 3 2 1	NO
3 1 3 5	YES



Задача Н. Ночной жор Даттона 7-8

Ограничение по времени: 1 секунда

У хомяка Даттона n переключателей, контролирующих свет. Он боится темноты, поэтому просит вас написать ему программу, которая поможет ему.

Его дом выглядит как коридор, состоящий из n комнат, соединенных в ряд. В каждой комнате есть лампочка, которая изначально может быть включена «1» или выключена «0».

Начальные состояния лампочек описаны массивом L .

Даттон часто просыпается, чтобы пойти на кухню, поэтому перед сном хочет переключить все лампочки так, чтобы он мог пройти по коридору из своей спальни на кухню так, чтобы в каждой комнате, лампочка была включена.

Кухня находится в последней комнате, то есть в комнатах с номерами с 1 по n лампочки должны быть включены.

Переключать лампочки он может следующим образом: выбрать k подряд идущих комнат коридора, во всех комнатах из k выбранных, если лампочки были включены, они станут выключенными, и наоборот. То есть «0» будут изменяться на «1», а «1» на «0».

Даттон устал после тяжелого рабочего дня, поэтому хочет сделать как можно меньше таких переключений.

Ваша задача — найти минимальное количество действий, которые Даттону нужно сделать или сказать, что его просьбу выполнить невозможно. Если при любом количестве действий и любой их последовательности так сделать невозможно, то выведите -1 .

Даны четыре тестовых массива и числа k , для которых требуется найти минимальное количество действий. Если вы не знаете ответ для теста или невозможно включить все лампочки, то запишите -1 .

Как ответ на эту задачу прикрепите **txt** файл. Отсылаете **txt** файл то на первой строчке будет записан ответ для первого теста, во второй строчке будет ответ для второго теста и тд.

По этой задаче на проверку принимается не более трех файлов.

Каждый тест оценивается в 25 баллов.

Номер примера	Значение k	Массив L	Ответ
1	2	[0, 0, 1, 1, 0, 1, 0]	(поле ответа)
2	3	[1, 0, 0, 1, 0, 1, 1, 1, 0, 0]	(поле ответа)
3	3	[1, 1, 1, 0, 1, 1, 1, 0, 0, 1]	(поле ответа)
4	7	[1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 1]	(поле ответа)



Задача I. Прimitивный тетрис

Ограничение по времени: 2 секунды

Как известно, игра тетрис проходит на клеточном поле размера $n \times m$ (n строк и m столбцов), на котором каждую единицу времени с верхнего края появляются клеточные фигуры нескольких видов и начинают двигаться с фиксированной скоростью вниз.

Задача игрока — поворачивать или перемещать эти фигуры так, чтобы наиболее равномерно заполнять это поле. В оригинальной игре как только строка поля оказывается целиком заполнена клетками фигур, она удаляется, поэтому игра может длиться бесконечно.

В нашей же версии никакие клетки удаляться не будут, и игра завершится как только верхняя клетка какой-то фигуры выйдет за верхнюю границу поля. Также от вас не требуется реализовывать стратегию игрока — достаточно по последовательности команд вида «поместить фигуру F_i в столбец j_i » сообщить, после какого действия игра завершится, и как будет выглядеть поле.

Каждая фигура состоит из четырех клеток, связанных по стороне (стандартные фигуры тетриса) и задается «рисунком» из символов ‘.’ (пустая клетка) и ‘#’ (клетка фигуры). Например,

```
##          .#          .##          #####          ###  
##          ##          ##.          #####          ..#  
          .#          .#          .#          .#          .#
```

Если фигура F_i занимает несколько столбцов, j_i будет указывать на столбец игрового поля, занимаемый самой левой клеткой F_i . Каждая фигура движется вниз по полю до момента соприкосновения с другой фигурой или нижней границей поля, после чего останавливается и больше никогда не движется.

Изначально поле пустое. Определите номер фигуры, на которой игра закончится, и как будет выглядеть поле после в конце игры.

Формат входных данных

В первой строке даны три целых числа n , m и q — размеры игрового поля и общее количество фигур, которые надо обработать ($1 \leq n \leq 10^4$; $4 \leq m \leq 50$; $1 \leq q \leq 4 \cdot 10^5$).

Далее следуют описания q фигур.

Первая строка описания фигуры содержит целые числа r_i , c_i , j_i — размеры содержащего фигуру поля и номер столбца игрового поля, на котором эта фигура появится ($1 \leq r_i, c_i \leq 4$; $1 \leq j_i \leq m - c_i + 1$).

Затем следуют r_i строк по c_i символов ‘.’ или ‘#’ в каждой, описывающие фигуру. Гарантируется, что заданная таким образом фигура связна и состоит ровно из четырех клеток (то есть является корректной фигурой тетриса).

Формат выходных данных

В первой строке выведите номер первой фигуры (от 1 до q), которая выйдет за верхнюю границу поля. Если все q фигур поместятся на поле, выведите $q + 1$.

Затем выведите n строк по m символов ‘.’ или ‘#’ в каждой — состояние поля на момент окончания игры. Если последняя фигура вышла за верхнюю границу поля, ее часть за границей поля выводить не надо.

Система оценки

Каждая группа тестов будет оцениваться только если предварительно были пройдены необходимые группы тестов, и баллы начисляются в случае, если все тесты группы пройдены. Тесты из условия не оцениваются. Все тесты разбиты на группы со следующими ограничениями:



Подзадача	Доп. ограничения	Необх. подзадачи	Баллы
1	$n = 1; m = 4$		5
2	$n \leq 3; m = 4$	1	5
3	$c_i = 1$ для всех i		8
4	$m \bmod 2 = 0;$ $c_i = r_i = 2$ и $j_i \bmod 2 = 0$ для всех i		13
5	$r_i = 1$ для всех i		11
6	$n, m \leq 15; q \leq 50$	1, 2	16
7	$n, q \leq 800$	1, 2, 6	19
8	нет	1 – 7	23

Пример

стандартный ввод	стандартный вывод
3 5 4 1 4 1 #### 3 2 4 #. ## . # 2 2 1 ## ## 2 3 2 ### . #.	4 ####. ##. ## #####



Задача J. Прimitивный тетрис 7-8

Ограничение по времени: 2 секунды

Как известно, игра тетрис проходит на клеточном поле размера $n \times m$ (n строк и m столбцов), на котором каждую единицу времени с верхнего края появляются клеточные фигуры нескольких видов и начинают двигаться с фиксированной скоростью вниз.

Задача игрока — поворачивать или перемещать эти фигуры так, чтобы наиболее равномерно заполнять это поле. В оригинальной игре как только строка поля оказывается целиком заполнена клетками фигур, она удаляется, поэтому игра может длиться бесконечно.

В нашей же версии никакие клетки удаляться не будут, и игра завершится как только верхняя клетка какой-то фигуры выйдет за верхнюю границу поля. Также от вас не требуется реализовывать стратегию игрока — достаточно по последовательности команд вида «поместить фигуру F_i в столбец j_i » определить, как будет выглядеть поле в конце игры.

Каждая фигура состоит из четырех клеток, связанных по стороне (стандартные фигуры тетриса) и задается «рисунком» из символов ‘.’ (пустая клетка) и ‘#’ (клетка фигуры). Например,

```
##          .#          .##          #####          ###  
##          ##          ##.          #####          ..#  
          .#          .#          .#          .#          .#
```

Если фигура F_i занимает несколько столбцов, j_i будет указывать на столбец игрового поля, занимаемый самой левой клеткой F_i . Каждая фигура движется вниз по полю до момента соприкосновения с другой фигурой или нижней границей поля, после чего останавливается и больше никогда не движется.

Изначально поле пустое. Определите, как будет выглядеть поле после в конце игры.

Как ответ на эту задачу прикрепите `txt` файл. Отсылаете `txt` файл то на первой свободной строчке будет записан ответ для первого теста, во второй свободной строчке будет ответ для второго теста и тд.

По этой задаче на проверку принимается не более пяти файлов.

Формат входных данных

В первой строке входных данных дано число записано число 4 — количество входных тестов.

Далее следуют описание тестов. В первой строке даны три целых числа n , m и q — размеры игрового поля и общее количество фигур, которые надо обработать ($1 \leq n \leq 10^4$; $4 \leq m \leq 50$; $1 \leq q \leq 4 \cdot 10^5$).

Далее следуют описания q фигур.

Первая строка описания фигуры содержит целые числа r_i , c_i , j_i — размеры содержащего фигуру поля и номер столбца игрового поля, на котором эта фигура появится ($1 \leq r_i, c_i \leq 4$; $1 \leq j_i \leq m - c_i + 1$).

Затем следуют r_i строк по c_i символов ‘.’ или ‘#’ в каждой, описывающие фигуру. Гарантируется, что заданная таким образом фигура связна и состоит ровно из четырех клеток (то есть является корректной фигурой тетриса).

Формат выходных данных

Выведите на каждый тест n строк по m символов ‘.’ или ‘#’ в каждой — состояние поля на момент окончания игры. Если последняя фигура вышла за верхнюю границу поля, ее часть за границей поля выводить не надо.

Система оценки

В задаче четыре теста, каждый оценивается в 25 баллов. За каждый тест вы получите $\max(0, 25 - 2X)$ баллов, где X — число клеток, в которых ваш ответ отличается от правильного ответа.